Visualizing Classifications of Hierarchical Models of Cortex

Will Landecker[1], Steven P. Brumby[2], Mick Thomure[1], Cristina Rinaudo[2], Garrett T. Kenyon[2], Luis M.A. Bettencourt[2,3] and Melanie Mitchell[1,3]

1. Portland State University, USA
2. Los Alamos National Laboratory, USA
3. Santa Fe Institute, USA

Correspondence:
Will Landecker: landeckw@cs.pdx.edu

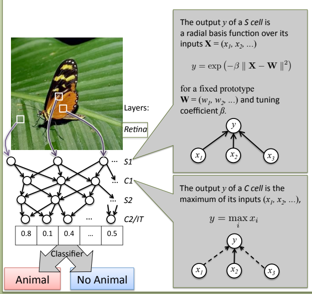
1. Feed-forward hierarchical models of cortex

The HMAX model of visual cortex (an alternating network of *S cells* and *C cells* based on the simple and complex cells of Hubel and Wiesel [1962]) achieves accuracy above 80% in the complex task of detecting animals in images of natural scenery [Serre et al., 2007]. However, it is unclear whether the model is using features extracted from the animal or finding spurious statistics in the data set [Pinto et al., 2008]. We implement an HMAX model (PANN: Petascale Artificial Neural Network [Brumby et al., 2009]) and propose a method of answering this question,

Which image regions caused the classification of the image?

Results indicate that PANN's detection of animals with an unbiased linear-kernel SVM (Support Vector Machine) is sometimes based on the image background rather than the animal itself.

Overview of HMAX model



Form of the classifier

Given an input $\mathbf{x} = (x_1, x_2, ..., x_n)$, we define the additive classification function

$$\hat{y} = \mathrm{sgn}\left[\sum_{i=1}^{n} f_i(x_i)\right]$$

where we call $f_i(x_i)$ the *contribution* of feature *i* [Poulin et al. 2006]. For a Naïve Bayes classifier, we have

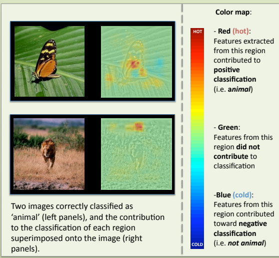
$$f_i(x_i) = \ln \frac{P(x_i|\text{class} = +)}{P(x_i|\text{class} = -)}$$

For an *unbiased* linear-kernel Support Vector Machine (SVM) with support vectors $\mathbf{w} = (w_1, w_2, ..., w_n) \in V$ and coefficients $\alpha_{\mathbf{w}} \in \mathbb{R}$ we have

$$f_i(x_i) = \sum_{\mathbf{w} \in V} \alpha_{\mathbf{w}} w_i x_i$$

In our experiments, unbiased SVMs performed as well as biased SVMs, in which the classification function includes a constant offset.

2. Visualizing the classification



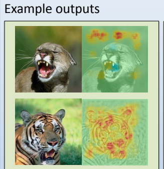
Two images correctly classified as 'animal' (left panels), and the contribution to the classification of each region superimposed onto the image (right panels).

We created an algorithm for visualizing the degree to which different regions contributed to the image's classification. In the right panels above, the color of a region indicates whether the image region contributed features that pulled the image's classification toward the positive (red) or negative (blue) class. In particular,

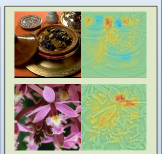
The 'hotter' regions caused the image to be classified as 'animal'.

Above, we see that the 'hot' regions are sometimes a part of the animal (top row), and sometimes they are not (bottom row).

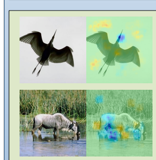
Example outputs



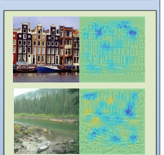
True Positive (classified as *animal*)



False Positive (classified as *animal*)



False Negative (classified as *not animal*)

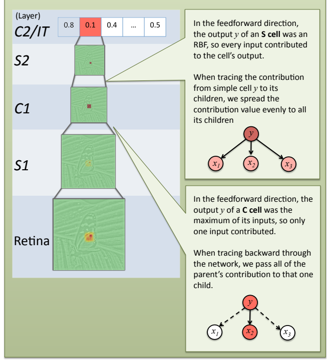


True Negative (classified as *not animal*)

3. Tracing classification decision through the network

In order to visualize the classification of an image, we trace the contribution $f_i(x_i)$ of each feature i. This process begins at the feature vector (the output of the model), and traces back to the image (the input to the model).

We illustrate this process for a single feature below.



Equations for tracing the classification

For an additive classifier, the i^{th} feature contributes the value $f_i(x_i)$ to the classification function; we trace this contribution down through the *S* and *C* cells to determine the contribution of each image region in the classification of the entire image.

S Cells

The feedforward output of a simple cell with inputs $\mathbf{X} = (x_1, x_2, ..., x_N)$ was determined by the RBF

$$y = \exp\left(-\beta \parallel \mathbf{X} - \mathbf{W} \parallel^2\right)$$

where $\mathbf{W} = (w_1, w_2, ..., w_N)$ is fixed for the cell.

Because each input contributed to the simple cell's output, we divide the cell's contribution evenly among its inputs. Thus if the simple cell's contribution was H_S then we define the contribution of each input as

$$\frac{H_S}{N}$$

C Cells

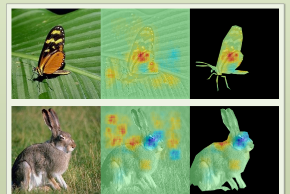
The feedforward output of a complex cell with inputs $\mathbf{X} = (x_1, x_2, ..., x_N)$ was determined by the max operation

$$y = \max_i x_i$$

Because only a single input contributed to the complex cell's output, we distribute the cell's contribution completely to that one input. Thus if the complex cell's contribution was H_C then the contribution of input x_i is defined as

$$H_{x_i} = \begin{cases} H_C & \text{if } x_i = \max_j x_j \\ 0 & \text{otherwise} \end{cases}$$

4. Was the classification based on the animal or the background?



Examples of images (left), the contribution of each image region to the classification (center), and the contribution of only the regions that overlap the animal, using hand-segmented images (right).

Our visualization shows which image regions contributed to the image's classification. Clearly, the correct classification of *animal* is sometimes caused by the image's background. This inspires the question,

What would the class be if we considered only the contributions of features extracted from pixels belonging to the animal?

Using hand-segmented images (above), we annotate which image regions belong to the object, and which belong to the background. This allows us to reclassify the image using only features extracted from the animal, and not from the background.

Reclassification using features from the animal

We partition the features into those that were extracted from the animal ($\mathcal{A}$), and those that were extracted from the background ($\mathcal{B}$). This allows us to rewrite our classification function,

$$\hat{y} = \mathrm{sgn}\left[\sum_{i=1}^{n} f_i(x_i)\right] = \mathrm{sgn}\left[\sum_{i \in \mathcal{A}} f_i(x_i) + \sum_{j \in \mathcal{B}} f_j(x_j)\right]$$

Now we reclassify using only the features extracted from the object,

$$\hat{y}_{\mathcal{A}} := \mathrm{sgn}\left[\sum_{i \in \mathcal{A}} f_i(x_i)\right]$$

Comparing our new class prediction $\hat{y}_{\mathcal{A}}$ to our previous prediction $\hat{y}$ tells us whether the correct classification of *animal* was based on features extracted from the animal or from the background of the image.

5. Experiment

We tested our model on the binary decision task of determining whether or not an animal appears in an image of natural scenery. Using the AnimalDB data set [Torralba and Oliva, 2003], we trained our model with 600 images and tested on the remaining 600.

We compared the classification accuracy of both Naïve Bayes and unbiased linear-kernel SVM classifiers on the test set, considering all features extracted from the image (*'full image'*) as well as features only extracted from regions of the image containing the animal (*'animal only'*).

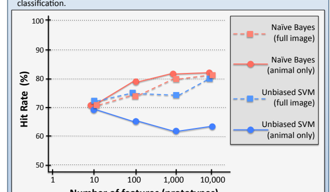
Classifier	Features Used	Hit Rate	False Alarm Rate	Test Accuracy	d'
Naïve Bayes	Full image	82.2%	37.9%	72.3%	1.23
Naïve Bayes	Animal only	83.3%	37.9%	72.7%	1.27
Linear-kernel SVM	Full image	80.8%	21.8%	79.5%	1.65
Linear-kernel SVM	Animal only	65.0%	21.8%	71.6%	1.16

In the above table, we see that excluding the background features slightly improves Naïve Bayes, but it causes the SVM hit rate to decrease by 15.8%.

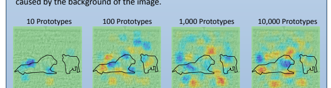
These results indicate that the SVM sometimes relies on features from the image's background when detecting an animal.

Sensitivity to number of features

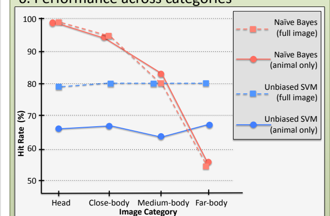
In our second experiment, we investigate how changing the number of features (prototypes) affects the role that the image's background plays during classification.



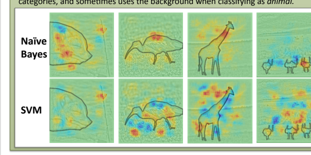
Although not plotted above, the *accuracy* of the SVM steadily increases from 64% (10 features) to 79% (10,000 features). The above graph indicates that a significant portion of this increase in accuracy is due to the SVM's classification of animals caused by the background of the image.



6. Performance across categories



The AnimalDB data set is organized into four distinct categories, based on the scale of the animal in the picture. Above, Naïve Bayes is sensitive to the image category, but is unaffected by the image backgrounds. SVM performs similarly for all image categories, and sometimes uses the background when classifying as *animal.*



Conclusions

- We have proposed a **new method for visualizing the classification decisions** of hierarchical feed-forward network models of object recognition.
- Our method **can be applied to any such model that uses an additive classifier** (e.g., SVM and Naïve Bayes).
- In particular, our method has shown that, for one implementation (PANN) of a standard visual cortex model (HMAX), **classifications are sometimes based on features of the background rather than the object to be recognized.**
- In general, **our method can reveal unintended correlations in image data sets as well as unintended behavior of the visual model and classifier.**

Future Work

- We plan to add **feedback methods for spatial attention and saliency detection** to PANN. Several methods for this have been proposed, including Bayesian networks [Chikkerur et al., 2009]. Such methods should have advantages in learning and using discriminating feature sets that are more relevant to the task.

References

D. Hubel and T. Wiesel, *Receptive Fields, Binocular Interaction and Functional Architecture in the Cat's Visual Cortex*. J. Physiol. **160**, 1962.
T. Serre, A. Oliva and T. Poggio. *A feedforward architecture accounts for rapid categorization*. Proceedings of the National Academy of Science, **104**(15), pp. 6424-6429, April 2007.
N. Pinto, D. Cox and J. DiCarlo, *Why is real-world visual object recognition hard?* PLoS Computational Biology, **4**(1), 2008.
S.P. Brumby et al., *Large-scale functional models of visual cortex for remote sensing*. 38th IEEE Applied Imagery Pattern Recognition, 2009
B. Poulin et al., *Visual explanation of evidence in additive classifiers*. Proceedings of 18th Conference on Innovative Applications of Artificial Intelligence, July 2006.
A. Torralba and A. Oliva, *Statistics of natural image categories.* Network: Computation in Neural Systems, **14,** 2003.
S. Chikkerur, T. Serre and T. Poggio, *A Bayesian inference theory of attention: neuroscience and algorithms*. MIT-CSAIL-TR-20009-047, 2009.
Work supported by Department of Energy (LDRD-DR-2009006) and National Science Foundation (Award No. NSF-OCI-0749348).